CLAIMS:

1. A method comprising:

responsive to a plurality of inputs, each input being defined by a first set of bits and a second set of at least one bit, for each input of the plurality of inputs and in parallel with other inputs of the plurality of inputs:

for each of a plurality of look-up tables each having a plurality of elements, looking-up one of the plurality of elements of the look-up table using the first set of bits that define the input to obtain an output, the output from each of the plurality of look-up tables collectively comprising a set of corresponding outputs; and

selecting a corresponding output from the set of corresponding outputs using the second set of at least one bit that defines the input.

2. A method according to claim 1 wherein the plurality of elements of each look-up table collectively comprise a combined table of elements each having a pre-determined value obtained using an S7 function.

3. A method according to claim 1 wherein for each look-up table, the plurality of elements of the look-up table and the plurality of inputs are loaded as vectors and the looking-up comprises for each of the inputs selecting one of the plurality of elements of the look-up table using the first set of bits that define the input.

4. A method according to claim 3 comprising using a vperm (vector permutation) instruction for the selecting one of the plurality of elements of the look-up table using the first set of bits that define the input.

5. A method according to claim 1 wherein for each of the plurality of inputs, the second set of at least one bit that defines the input comprises one bit and the set of corresponding outputs comprises two corresponding outputs, and wherein for each of the plurality of inputs the selecting comprises:

selecting one of the two outputs using the one bit of the at least one bit that defines the input.

6. A method according to claim 1 wherein for each of the plurality of inputs, the second set of at least one bit that defines the input comprises at least two bits, and wherein for each of the plurality of inputs the selecting comprises:

successively performing a selection on a remaining number of corresponding outputs of the set of corresponding outputs for each bit of the at least two bits, the number of corresponding outputs remaining being equal to all of the corresponding outputs of the set of corresponding outputs a first time the selection is performed, the selection being replacing the remaining number of corresponding outputs with a selection of half of the remaining number of outputs using a respective bit of the at least two bits, the selection of half of the remaining number of outputs being the number of remaining outputs for the next time the selection is performed.

7. A method according to claim 6 wherein for each time the selection on a remaining number of corresponding outputs is performed, the remaining number of corresponding outputs comprises at least one set of two remaining corresponding outputs and the selection of half of the remaining number of outputs comprises, for each set of two corresponding outputs of the at least one set of two remaining corresponding outputs:

replicating the respective bit into a plurality of replicated bits; and

using a vector instruction, selecting one of the two remaining corresponding outputs depending on the plurality of replicated bits.

8. A method according to claim 7 wherein the vector instruction is a vsel (vector select instruction).

9. A method according to claim 2 wherein for each input, the first set of bits that define the input comprises five bits, the second set of bits that define the input comprises two bits and the look-up tables comprise four look-up tables, wherein for each of the four look-up tables the plurality of inputs and the plurality of elements of the look-up table are loaded as vectors and the looking-up comprises for each of the inputs selecting one of the plurality of elements of the look-up table using the first set of bits that define the input.

10. A method according to claim 2 wherein for each input, the first set of bits that define the input comprises four bits, the second set of bits that define the input comprises three bits and the look-up tables comprise eight look-up tables, and wherein for each of the eight look-up tables the plurality of inputs and the plurality of elements of the look-up table are loaded as vectors and for each of the inputs the looking-up comprises selecting one of the plurality of elements of the look-up table using the first set of bits that define the input.

11. A method according to claim 2 applied in ciphering data in a Kasumi implementation.

12. An apparatus comprising:

a memory adapted to store a plurality of elements of each of a plurality of look-up tables; and

a processor adapted to:

responsive to receiving a plurality of inputs, each input being defined by a first set of bits and a second set of at least one bit, for each input of the plurality of inputs and in parallel with other inputs of the plurality of inputs:

for each of the plurality of look-up tables, look-up one of the plurality of elements of the look-up table using the first set of bits that define the input to obtain an output, the output from each of the plurality of look-up tables collectively comprising a set of corresponding outputs; and

select a corresponding output from the set of corresponding outputs using the second set of at least one bit that define the input.

13. An apparatus according to claim 12 wherein the plurality of elements of each look-up table collectively comprise a combined table of elements each having a pre-determined value obtained using an S7 function.

14. An apparatus according to claim 12 wherein for each look-up table, the plurality of elements of the look-up table and the plurality of inputs are loaded as vectors and for each of the inputs the processor is further adapted to select one of the plurality of elements of the look-up table using the first set of bits that define the input.

15. An apparatus according to claim 14 the processor comprises an Altivec co-processor having a vperm (vector permutation) instruction, the processor being adapted to use the vperm instruction for the selecting one of the plurality of

elements of the look-up table using the first set of bits that define the input.

16. An apparatus according to claim 12 wherein for each of the plurality of inputs, the second set of at least one bit that defines the input comprises at least two bits, and wherein for each of the plurality of inputs in selecting the corresponding output from the set of corresponding outputs the processor is adapted to:

successively perform a selection on a remaining number of corresponding outputs of the set of corresponding outputs for each bit of the at least two bits, the number of corresponding outputs remaining being equal to all of the corresponding outputs of the set of corresponding outputs a first time the selection is performed, the selection being replacing the remaining number of corresponding outputs with a selection of half of the remaining number of outputs using a respective bit of the at least two bits, the selection of half of the remaining number of outputs being the number of remaining outputs for the next time the selection is performed.

17. An apparatus according to claim 16 wherein for each time the selection on a remaining number of corresponding outputs is performed, the remaining number of corresponding outputs comprises at least one set of two remaining corresponding outputs and the selection of half of the remaining number of outputs comprises, for each set of two corresponding outputs of the at least one set of two remaining corresponding outputs the processor being adapted to:

replicate the respective bit into a plurality of replicated bits; and

using a vector instruction, select one of the two remaining corresponding outputs depending on the plurality of replicated bits.

18. An apparatus according to claim 17 wherein the processor comprises an Altivec co-processor having a vsel (vector select instruction), the vsel instruction being the vector instruction.

19. An apparatus according to claim 13 wherein for each input, the first set of bits that define the input comprises five bits, the second set of bits that define the input comprises two bits and the look-up tables comprise four look-up tables, wherein for each of the four look-up tables the plurality of inputs and the plurality of elements of the look-up table are loaded as vectors and for each of the inputs the processor is adapted to select one of the plurality of elements of the look-up table using the first set of bits that define the input.

20. An apparatus according to claim 13 wherein for each input, the first set of bits that define the input comprises four bits, the second set of bits that define the input comprises three bits and the look-up tables comprise eight look-up tables, and wherein for each of the eight look-up tables the plurality of inputs and the plurality of elements of the look-up table are loaded as vectors and for each of the inputs the processor is adapted to select one of the plurality of elements of the look-up table using the first set of bits that define the input.

21. A method comprising:

responsive to a plurality of inputs each defined by a first plurality of bits, for each input of the plurality of

inputs and in parallel with other inputs of the plurality of inputs:

for each of a plurality of look-up tables each having a plurality of elements:

selecting a respective subset of bits of the first plurality of bits that define the input, the bits of the respective subset of bits comprising fewer bits than the first plurality of bits of the input; and

looking-up an element of the plurality of elements of the look-up table using the subset of bits to obtain an output; and

combining the outputs obtained from the plurality of look-up tables to obtain at least one bit.

22. A method according to claim 21 wherein for each input of the plurality of inputs, the outputs obtained from the plurality of look-up tables each comprise a second plurality of bits, the second plurality of bits comprising fewer bits than the first plurality of bits of the input.

23. A method according to claim 22 wherein for each input of the plurality of inputs, the at least one bit comprises a third plurality of bits, the third plurality of bits comprising the same number of bits as the first plurality of bits of the input.

24. A method according to claim 21 wherein for at least one look-up table of the plurality of look-up tables, for each input the selecting comprises manipulating at least one of the plurality of bits that define the input using at least one of a bit rotation instruction and a bit shifting instruction.

25. A method according to claim 24 wherein for each of the at least one look-up table, for each input the manipulating at least one of the first plurality of bits comprises ordering the respective subset of bits of the input as least significant bits.

26. A method according to claim 23 wherein each element of the plurality of elements of each look-up table has a pre-determined value.

27. A method according to claim 26 wherein for each input of the plurality of inputs the first plurality of bits and the third plurality of bits each comprise 9 bits, the pre-determined value of each of the plurality of elements of each of the plurality of look-up tables is obtained from a partial evaluation of an S9 function.

28. A method according to claim 27 wherein for each look-up table of the plurality of look-up tables, the pre-determined value of each of the plurality of elements of the look-up table is a function of a number being definable by a bit sequence of one of 4 and 5 bits.

29. A method according to claim 28 wherein for each input of the plurality of inputs, for each look-up table the respective subset of bits of the first plurality of bits that define the input comprises one of 4 and 5 bits and the look-up table is looked-up using a vperm (vector permutation) instruction.

30. A method according to claim 27 wherein for each input of the plurality of inputs, the combining comprises performing a plurality of exclusive-OR operations on the outputs obtained from the plurality of look-up tables for the input.

31. A method according to claim 30 wherein for each input of the plurality of inputs, the combining comprises manipulating the second plurality of bits of at least one output of the outputs obtained from the plurality of look-up tables for the input using one of a bit shifting instruction and a bit rotation instruction.

32. A method according to claim 31 wherein the bit shifting instruction comprises one of a vector shift right byte instruction and a vector shift left byte instruction and the bit rotation instruction comprises one of a vector rotate left byte instruction and a vector rotate right byte instruction.

33. A method according to claim 30 wherein for each input of the plurality of inputs, the combining comprises:

for a first output of the outputs obtained from the plurality of look-up tables for the input, manipulating the second plurality of bits of the first output using one of a bit rotation instruction and a bit shifting instruction; and

for a second output of the outputs obtained from the plurality of look-up tables for the input, performing one of the plurality of exclusive-OR operations on the second output and the first output to obtain a third output having a fourth plurality of bits.

34. A method according to claim 30 wherein for each input, the bits of the second plurality of bits of each respective subset of bits of the first plurality of bits of the input have a pre-determined order and are each used for obtaining a respective one of the third plurality of bits, the outputs obtained from the look-up tables collectively comprising at least one group of outputs each having at least two outputs of the outputs obtained from the look-up tables,

for each group of outputs of the at least one group of outputs the at least two outputs in the group of outputs having bits used for determining a common subset of bits of the third plurality of bits, the combining comprising:

for each group of outputs of the at least of group of outputs, combining the at least two outputs of the group of outputs using at least one of the plurality of exclusive-OR operations.

35. An apparatus comprising:

a memory adapted to store a plurality of elements of each of a plurality of look-up tables; and

a processor adapted to:

responsive to a plurality of inputs each defined by a first plurality of bits, for each input of the plurality of inputs and in parallel with other inputs of the plurality of inputs:

for each look-up table of the plurality of look-up tables:

select a respective subset of bits of the first plurality of bits that define the input, the bits of the respective subset of bits comprising fewer bits than the first plurality of bits of the input; and

look-up an element of the plurality of elements of the look-up table using the subset of bits to obtain an output; and

combine the outputs obtained from the plurality of look-up tables to obtain at least one bit.

36. An apparatus according to claim 35 wherein for each input of the plurality of inputs, the outputs obtained from the plurality of look-up tables each comprise a second plurality of bits, the second plurality of bits comprising fewer bits than the first plurality of bits of the input.

37. An apparatus according to claim 36 wherein for each input of the plurality of inputs, the at least one bit comprises a third plurality of bits, the third plurality of bits comprising the same number of bits as the first plurality of bits of the input.

38. An apparatus according to claim 35 wherein for at least one look-up table of the plurality of look-up tables, for each input the processor is adapted to manipulate at least one of the first plurality of bits that define the input using at least one of a bit rotation instruction and a bit shifting instruction.

39. An apparatus according to claim 38 wherein for each of the at least one look-up table:

for each input the processor is adapted to manipulate the at least one of the first plurality of bits by ordering the respective subset of bits of the input as least significant bits.

40. An apparatus according to claim 37 wherein each element of the plurality of elements of each look-up table has a pre-determined value.

41. An apparatus according to claim 40 wherein for each input of the plurality of inputs the first plurality of bits and the third plurality of bits each comprise 9 bits, the pre-determined value of each of the plurality of elements of each

of the plurality of look-up tables is obtained from a partial evaluation of an S9 function.

42. An apparatus according to claim 41 wherein for each look-up table of the plurality of look-up tables, the pre-determined value of each of the plurality of elements of the look-up table is a function of a number being definable by a bit sequence of one of 4 and 5 bits.

43. An apparatus according to claim 42 wherein for each input of the plurality of inputs, for each look-up table the respective subset of bits of the first plurality of bits that define the input comprises one of 4 and 5 bits, the processor being adapted to look-up the look-up table using a vperm (vector permutation) instruction.

44. An apparatus according to claim 41 wherein for each input of the plurality of inputs, the processor is adapted to perform a plurality of exclusive-OR operations on the outputs obtained from the plurality of look-up tables for the input.

45. An apparatus according to claim 44 wherein for each input of the plurality of inputs, the processor is adapted to manipulate the second plurality of bits of at least one output of the outputs using one of a bit shifting instruction and bit rotation instruction.

46. A method according to claim 45 wherein the bit shifting instruction comprises one of a vector shift right byte instruction and a vector shift left byte instruction and the bit rotation instruction comprises one of a vector rotate left byte instruction and a vector rotate right byte instruction.

47. An apparatus according to claim 44 wherein for each input of the plurality of inputs, the processor is adapted to:

for a first output of the outputs obtained from the plurality of look-up tables for the input, manipulate the second plurality of bits of the first output using one of a bit rotation instruction and a bit shifting instruction; and

for a second output of the outputs obtained from the plurality of look-up tables for the input, perform one of the plurality of exclusive-OR operations on the second output and the first output to obtain a third output having a fourth plurality of bits.

48. An apparatus according to claim 44 wherein for each input, the bits of the second plurality of bits of each respective subset of bits of the first plurality of bits of the input have a pre-determined order and are each used for obtaining a respective one of the third plurality of bits, the outputs obtained from the look-up tables collectively comprising at least one group of outputs each having at least two outputs of the outputs obtained from the look-up tables, for each group of outputs of the at least one group of outputs the at least two outputs in the group of outputs having bits used for determining a common subset of bits of the third plurality of bits, the processor being adapted to:

for each group of outputs of the at least of group of outputs, combine the at least two outputs of the group of outputs using at least one of the plurality of exclusive-OR operations.

49. An article of manufacture comprising:

a computer usable medium having computer readable program code means embodied therein, the computer readable code means in said article of manufacture comprising:

responsive to a plurality of inputs, each input being defined by a first set of bits and a second set of at least one bit, for each input of the plurality of inputs and in parallel with other inputs of the plurality of inputs;

computer readable code means for, for each of a plurality of look-up tables each having a plurality of elements, looking-up one of the plurality of elements of the look-up table using the first set of bits that define the input to obtain an output, the output from each of the plurality of look-up tables collectively comprising a set of corresponding outputs; and

computer readable code means for selecting a corresponding output from the set of corresponding outputs using the second set of at least one bit that defines the input.

50. An article of manufacture comprising:

a computer usable medium having computer readable program code means embodied therein, the computer readable code means in said article of manufacture comprising:

responsive to a plurality of inputs each defined by a first plurality of bits, for each input of the plurality of inputs and in parallel with other inputs of the plurality of inputs:

computer readable code means for, for each of a plurality of look-up tables each having a plurality of elements:

selecting a respective subset of bits of the first plurality of bits that define the input, the bits of the

respective subset of bits comprising fewer bits than the first plurality of bits of the input; and

looking-up an element of the plurality of elements of the look-up table using the subset of bits to obtain an output; and

computer readable code means for combining the outputs obtained from each look-up table to obtain at least one bit.

51. A method comprising:

responsive to N K_{in}-bit inputs:

performing bit permutation/reordering on the N K_{in}-bit inputs to produce M parallel sets of outputs wherein N and K_{in} are integers satisfying N, $K_{in} \geq 2$, an ith set of outputs of the M parallel sets of outputs containing N sets of bits $L_{i,in}$ bits in length with i and $L_{i,in}$ being integers satisfying i = 1 to M and $1 \leq L_{i,in} < K_{in}$, the ith set of outputs defining a respective subset of the K_{in} bits of the inputs;

for each parallel set of outputs, performing a parallel lookup table operation to generate a corresponding parallel set of outputs containing N outputs, each being associated with a respective one of the N K_{in}-bit inputs and each being $L_{i,out}$ bits in length, $L_{i,out}$ being an integer satisfying $L_{i,out} \geq 1$; and

for each of the N K_{in}-bit inputs, generating a respective output by performing a bit combining operation on the outputs from the parallel look-up table operations associated with the input.

52. A method according to claim 51 wherein for each of the N K_{in}-bit inputs, the generating comprises performing a bit manipulation on the outputs of the parallel look-up table operations associated with the input.

53. A method according to claim 51 wherein the bit combining operations are implemented in parallel.

54. A method according to claim 51 wherein for each of the N K_{in}-bit inputs the respective output generated comprises K_{out} bits, K_{out} being an integer satisfying $K_{out} \geq 1$, and wherein in performing the bit permutation/reordering on the N K_{in}-bit inputs, the ith set of outputs defining the respective subset of the K_{in} bits of the inputs is selected such that the respective subset of the K_{in} bits effects only a defined maximum number $Pi < K_{out}$ bits of the respective outputs wherein Pi is an integer.

55. A method of generating a plurality of outputs according to a ciphering algorithm which for each of the plurality of outputs operates on a respective input using a respective key, the ciphering algorithm comprising a plurality of rounds in which functions are evaluated, the method comprising, for at least one function of the functions of at least one of the plurality of rounds:

responsive to a plurality of first inputs each being associated with one of the respective inputs, for each first input and in parallel with other first inputs of the plurality of first inputs:

generating an output by looking up at least one look-up table using the input, each look-up table having a plurality of elements.

56. A method according to claim 55 wherein the ciphering algorithm is a Kasumi algorithm.

57. A method according to claim 55 wherein for a function of a certain type of the at least one function the at least one look-up table comprising a plurality of look-up tables and the output from each of the plurality of look-up tables collectively comprising a set of corresponding outputs, each first input of the plurality of first inputs being defined by a first set of bits and a second set of at least one bit, the method comprising for each first input of the plurality of first inputs and in parallel with the other first inputs of the plurality of first inputs:

selecting a corresponding output from the set of corresponding outputs using the second set of at least one bit that defines the input.

58. A method according to claim 57 wherein the ciphering algorithm is a Kasumi algorithm and the function of a certain type is an S7 function.

59. A method according to claim 55 wherein for a function of a certain type of the at least one function the at least one look-up table comprises a plurality of look-up tables and each first input of the plurality of first inputs is defined by a first plurality of bits, the method comprising:

for each first input of the plurality of first inputs and in parallel with the other first inputs of the plurality of first inputs:

for each of the plurality of look-up tables:

selecting a respective subset of bits of the first plurality of bits that define the first input, the

bits of the respective subset of bits comprising fewer bits than the first plurality of bits of the first input, the look-up table being looked up using the subset of bits to obtain the output; and

combining the outputs obtained from the plurality of look-up tables to obtain at least one bit.

60. A method according to claim 59 wherein the ciphering algorithm is a Kasumi algorithm and the function of a certain type is an S9 function.

61. A method according to claim 56 wherein the at least one round comprises the plurality of rounds and wherein for each round the at least one function comprises six S7 functions and six S9 functions, the method further comprising for each function of the plurality of functions other than the at least one function:

responsive to a plurality of second inputs each being associated with one of the respective inputs, and in parallel with other second inputs of the plurality of second inputs:

generating an output according to the function using the input.

62. A method according to claim 55 further comprising, for each output of the plurality of outputs and in parallel with other outputs of the plurality of outputs:

combining the output with input data to generate ciphered data.

63. A method according to claim 62 wherein the combining comprises performing an exclusive-OR operation.

64. An apparatus for generating a plurality of outputs according to a ciphering algorithm which for each of the plurality of outputs operates on a respective input using a respective key, the ciphering algorithm comprising a plurality of rounds in which functions are evaluated, the apparatus comprising:

a memory adapted to store a plurality of elements of each of at least one look-up table; and

a processor adapted to:

for at least one function of the functions of at least one of the plurality of rounds:

responsive to a plurality of first inputs each being associated with one of the respective inputs, for each first input and in parallel with other first inputs of the plurality of first inputs:

generate an output by looking up at least one look-up table using the input, each look-up table having a plurality of elements.

65. An apparatus according to claim 64 wherein the ciphering algorithm is a Kasumi algorithm.

66. An apparatus according to claim 64 wherein for a function of a certain type of the at least one function, the at least one look-up table comprises a plurality of look-up tables and the output from each of the plurality of look-up tables collectively comprising a set of corresponding outputs, each first input of the plurality of first inputs being defined by a first set of bits and a second set of at least one bit, the processor being further adapted to:

for each first input of the plurality of first inputs and in parallel with the other first inputs of the plurality of first inputs:

select a corresponding output from the set of corresponding outputs using the second set of at least one bit that defines the input.

67. An apparatus according to claim 66 wherein the ciphering algorithm is a Kasumi algorithm and the function of a certain type is an S7 function.

68. An apparatus according to claim 64 wherein for a function of a certain type of the at least one function, the at least one look-up table comprises a plurality of look-up tables and each first input of the plurality of first inputs is defined by a first plurality of bits, the processor being further adapted to:

for each first input of the plurality of first inputs and in parallel with the other first inputs of the plurality of first inputs:

for each of the plurality of look-up tables:

select a respective subset of bits of the first plurality of bits that define the first input, the bits of the respective subset of bits comprising fewer bits than the first plurality of bits of the first input, the look-up table being looked up using the subset of bits to obtain the output; and

combine the outputs obtained from the plurality of look-up tables to obtain at least one bit.

69. An apparatus according to claim 68 wherein the ciphering algorithm is a Kasumi algorithm and the function of a certain type is an S9 function.

70. An apparatus according to claim 65 wherein the at least one round comprises the plurality of rounds and wherein for each round the at least one function comprises six S7 functions and six S9 functions, the processor being further adapted to:

for each function of the plurality of functions other than the at least one function:

responsive to a plurality of second inputs each being associated with one of the respective inputs, and in parallel with other second inputs of the plurality of second inputs:

generate an output according to the function using the input.

71. An apparatus according to claim 64 wherein the processor is further adapted to:

for each output of the plurality of outputs and in parallel with other outputs of the plurality of outputs:

combine the output with input data to generate ciphered data.

72. An apparatus according to claim 71 wherein the processor is adapted to combine the output with the input data using an exclusive-OR operation.

73. An article of manufacture comprising:

a computer usable medium having computer readable program code means embodied therein for generating a plurality

of outputs according to a ciphering algorithm which for each of the plurality of outputs operates on a respective input using a respective key, the ciphering algorithm comprising a plurality of rounds in which functions are evaluated, the computer readable code means in said article of manufacture comprising:

computer readable code means for:

for at least one function of the functions of at least one of the plurality of rounds:

responsive to a plurality of first inputs each being associated with one of the respective inputs, for each first input and in parallel with other first inputs of the plurality of first inputs:

generating an output by looking up at least one look-up table using the input, each look-up table having a plurality of elements.

74. A method comprising:

responsive to a plurality of inputs, each input being defined by at least one bit, for each input of the plurality of inputs and in parallel with other inputs of the plurality of inputs:

looking-up a look-up table having a plurality of elements using the at least one bit that define the input to obtain an output.

75. An apparatus comprising:

a memory adapted to store a plurality of elements of a look-up table; and

a processor adapted to:

responsive to a plurality of inputs, each input being defined by at least one bit, for each input of the plurality of inputs and in parallel with other inputs of the plurality of inputs:

look-up the look-up table using the at least one bit that define the input to obtain an output.

76. An article of manufacture comprising:

a computer usable medium having computer readable program code means embodied therein, the computer readable code means in said article of manufacture comprising:

computer readable code means for, responsive to a plurality of inputs, each input being defined by at least one bit, for each input of the plurality of inputs and in parallel with other inputs of the plurality of inputs:

looking-up a look-up table having a plurality of elements using the at least one bit that define the input to obtain an output.